

# HP StorageWorks Storage System Scripting Utility reference

Part number: T3724-96003  
Seventh edition: May 2005



**Legal and notice information**

Copyright © 1998-2005 Hewlett-Packard Development Company, L.P.

Hewlett-Packard Company makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

This document contains proprietary information, which is protected by copyright. No part of this document may be photocopied, reproduced, or translated into another language without the prior written consent of Hewlett-Packard. The information is provided "as is" without warranty of any kind and is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

HP StorageWorks Storage System Scripting Utility reference

---

# Contents

Preface . . . . .	9
About this guide . . . . .	9
Intended audience . . . . .	9
Prerequisites . . . . .	9
Related documentation . . . . .	9
Document conventions and symbols . . . . .	10
HP technical support . . . . .	10
Service tools . . . . .	11
HP-authorized reseller . . . . .	11
Helpful web sites . . . . .	11
Providing feedback . . . . .	11
 1 Getting started . . . . .	 13
Getting started . . . . .	14
Security considerations . . . . .	15
ProLiant servers and Smartstart 7.2 . . . . .	15
Installing the utility . . . . .	15
Installation . . . . .	15
Starting the utility . . . . .	15
Syntax . . . . .	15
Examples . . . . .	16
Shortcut commands . . . . .	16
Understanding paths and naming conventions . . . . .	16
Virtual disk example . . . . .	16
Disk groups example . . . . .	17
Aliases . . . . .	17
Virtual disk example . . . . .	17
Disk group example . . . . .	17
SET example . . . . .	17
Add snapshot example . . . . .	17
Delete virtual disk example . . . . .	18
Formatting and output of returned data . . . . .	18
Replica resync with snapclones . . . . .	18
Creating snapclones . . . . .	18
Cross Vraid . . . . .	18
 2 Utility command reference . . . . .	 21
Command introduction . . . . .	21
Menu map . . . . .	22
ADD . . . . .	25
ADD CONTAINER . . . . .	26
Syntax . . . . .	26
Switches . . . . .	26
ADD COPY . . . . .	27
Syntax . . . . .	27
Switches . . . . .	27
Required Switch . . . . .	28
ADD DR_GROUP . . . . .	28
Syntax . . . . .	28
Switches . . . . .	28

Required Switches . . . . .	30
ADD DISK_GROUP . . . . .	30
Syntax . . . . .	30
Switches . . . . .	30
ADD FOLDER . . . . .	31
Syntax . . . . .	31
Switch . . . . .	32
ADD HOST . . . . .	32
Syntax . . . . .	32
Switches . . . . .	32
ADD LUN . . . . .	33
Syntax . . . . .	33
Switches . . . . .	33
Required switches . . . . .	33
Examples . . . . .	34
ADD SNAPSHOT . . . . .	34
Syntax . . . . .	34
Required switch . . . . .	34
Switches . . . . .	34
ADD SYSTEM . . . . .	35
Syntax . . . . .	35
Switches . . . . .	36
ADD VDISK . . . . .	36
Syntax . . . . .	37
Switches . . . . .	37
Required switch . . . . .	39
Switch defaults . . . . .	39
Examples . . . . .	39
CAPTURE (CONFIGURATION) . . . . .	39
Syntax . . . . .	39
Step 1 scripts . . . . .	40
Step 1A script . . . . .	40
Step 1B script . . . . .	40
Step 1C script . . . . .	40
Step 2 script . . . . .	40
Step 3 script . . . . .	41
Restoring configurations . . . . .	41
Example . . . . .	41
CAPTURE (VALIDATE) . . . . .	41
Syntax . . . . .	41
CAPTURE VALIDATE example . . . . .	41
User functionality for CAPTURE commands . . . . .	42
DELETE . . . . .	42
DELETE DR_GROUP . . . . .	42
Syntax . . . . .	42
Example . . . . .	42
DELETE DISK_GROUP . . . . .	42
Syntax . . . . .	42
Example . . . . .	42
DELETE FOLDER . . . . .	43
Syntax . . . . .	43
Examples . . . . .	43
DELETE HOST . . . . .	43
Syntax . . . . .	43
Example . . . . .	43
DELETE LUN . . . . .	43
Syntax . . . . .	43
Example . . . . .	43
DELETE SYSTEM . . . . .	43
Syntax . . . . .	44

Example	44
DELETE VDISK	44
Syntax	44
Switches	44
EMVERSION	45
Syntax	45
Switches	45
Example	45
EXERCISE_DISK	45
Syntax	45
Switches	45
START	46
STOP	46
SUMMARY	46
Examples	47
EXIT	47
Syntax	47
Example	47
FILE	47
Syntax	48
Examples	48
HELP	48
Syntax	48
Examples	48
LS	49
Examples	49
LS CONTAINER	49
Syntax	49
LS CONTROLLER	49
Syntax	49
LS DISK	50
Syntax	50
LS DISK_GROUP	50
Syntax	50
LS DR_GROUP	50
Syntax	50
LS DISKSHELF	50
Syntax	50
LS FOLDER	50
Syntax	50
LS HOST	50
Syntax	50
LS LUN	51
Syntax	51
LS MANAGER	51
Syntax	51
Switch	51
LS OPTIONS	51
Syntax	51
LS SNAPSHOT	51
Syntax	51
LS SYSTEM	51
Syntax	52
LS TIME	52
Syntax	52
LS VDISK	52
Syntax	52
LS WORLD_WIDE_NAME	52
Syntax	52
REDISCOVER	52

Syntax . . . . .	52
RESTART . . . . .	52
Syntax . . . . .	53
Switches . . . . .	53
ALL_PEERS . . . . .	53
NOALL_PEERS . . . . .	53
Examples . . . . .	53
SELECT . . . . .	53
SELECT MANAGER . . . . .	53
Syntax . . . . .	53
Switches . . . . .	53
Required switches . . . . .	54
Example . . . . .	54
SELECT SYSTEM . . . . .	54
Syntax . . . . .	54
Examples . . . . .	54
SET . . . . .	54
SET DR_GROUP . . . . .	54
Syntax . . . . .	54
Switches . . . . .	54
SET DISK . . . . .	57
Syntax . . . . .	57
Switches . . . . .	57
SET DISK_GROUP . . . . .	57
Syntax . . . . .	57
Switches . . . . .	57
SET FOLDER . . . . .	58
Syntax . . . . .	58
Switches . . . . .	58
Example . . . . .	59
SET HOST . . . . .	59
Syntax . . . . .	59
Examples . . . . .	60
SET CONTROLLER . . . . .	60
Syntax . . . . .	60
Switches . . . . .	60
SET DISKSHELF . . . . .	60
Syntax . . . . .	60
Switches . . . . .	60
SET OPTIONS . . . . .	61
Syntax . . . . .	61
Switches . . . . .	61
Example . . . . .	62
SET SYSTEM . . . . .	62
Syntax . . . . .	63
Switches . . . . .	63
Example . . . . .	63
SET VDISK . . . . .	63
Syntax . . . . .	63
Switches . . . . .	63
Example . . . . .	65
SHUTDOWN . . . . .	65
Syntax . . . . .	65
Switches . . . . .	65
Examples . . . . .	65

### 3 Configuration examples . . . . . 67

---

# Figures

1 Menu map (part A)	23
2 Menu map (part B)	24
3 Menu map (part C)	25

---

# Tables

1 Document conventions . . . . . 10



---

# Preface

## About this guide

This guide describes how to configure and use the HP StorageWorks Storage System Scripting Utility to manage and monitor HP StorageWorks Enterprise Virtual Arrays (EVAs) in your environment. It provides information about:

- Starting the utility
- Utility commands and syntax

## Intended audience

This guide is intended as a reference guide for HP Command View Enterprise Virtual Array (EVA) users and administrators.

## Prerequisites

Prerequisites for using this product include knowledge of:

- Storage area networks (SANs)
- SAN fabric configurations
- HP StorageWorks Enterprise Virtual Array (EVA)
- Host operating systems intended for array monitoring

This guide assumes you have installed HP Command View EVA on a supported management server.

## Related documentation

In addition to this guide, please refer to other documents for this product:

- *HP StorageWorks Command View EVA installation guide*
- *HP StorageWorks EVA software compatibility reference*
- *HP StorageWorks Command View EVA online help (accessible from the HP Command View EVA user interface)*
- *HP StorageWorks Enterprise Virtual Array user guide*
- *HP StorageWorks Continuous Access EVA administrator's guide*

You can find these documents on the following HP web sites:

<http://h18006.www1.hp.com/products/storage/software/cmdvieweva/index.html>

<http://h18006.www1.hp.com/products/storage/software/conaccesseva/index.html>

<http://h18006.www1.hp.com/storage/arraysystems.html>

# Document conventions and symbols

**Table 1 Document conventions**

Convention	Element
Medium blue text: Figure 1	Cross-reference links and e-mail addresses
Medium blue, underlined text ( <a href="http://www.hp.com">http://www.hp.com</a> )	Web site addresses
<b>Bold font</b>	<ul style="list-style-type: none"><li>• Key names</li><li>• Text typed into a GUI element</li><li>• GUI elements that are clicked or selected, such as menu and list items, buttons, and check boxes</li></ul>
<i>Italics font</i>	Text emphasis
Monospace font or CAPITAL FONT	<ul style="list-style-type: none"><li>• File and directory names</li><li>• System output</li><li>• Code</li><li>• Commands</li></ul>
Monospace, italic font	<ul style="list-style-type: none"><li>• Code variables</li><li>• Command line variables</li></ul>
Monospace, bold font	Emphasis of file and directory names, system output, code, and commands



## CAUTION:

Indicates that failure to follow directions could result in damage to equipment or data.



## NOTE:

Provides additional information.

## HP technical support

Telephone numbers for worldwide technical support are listed on the HP web site:

<http://www.hp.com/support/>.

Collect the following information before calling:

- Technical support registration number (if applicable)
- Product serial numbers
- Product model names and numbers
- Applicable error messages
- Operating system type and revision level
- Detailed, specific questions

For continuous quality improvement, calls may be recorded or monitored.

HP strongly recommends that customers sign up online using the Subscriber's choice web site:

<http://www.hp.com/go/e-updates>.

- Subscribing to this service provides you with e-mail updates on the latest product enhancements, newest versions of drivers, and firmware documentation updates as well as instant access to numerous other product resources.
- After signing up, you can quickly locate your products by selecting **Business support** and then **Storage** under Product Category.

## Service tools

HP is constantly enhancing its service tools to keep pace with new product introductions. It is imperative that you check the service tools web site for the latest available downloads and install them on your system. This will ensure that the system tools work with your HP products effectively and efficiently. You can locate the services tools at the following web site: <http://h18000.www1.hp.com/support/svctools/>.

## HP-authorized reseller

For the name of your nearest HP-authorized reseller:

- In the United States, call 1-800-282-6672.
- Elsewhere, visit the HP web site: <http://www.hp.com>. Then click **Contact HP** to find locations and telephone numbers.

## Helpful web sites

For other product information, see the following web sites:

- <http://www.hp.com>
- <http://www.hp.com/go/storage>
- <http://www.hp.com/support/>
- <http://www.docs.hp.com>

## Providing feedback

To provide e-mail feedback on:

- HP Command View EVA: CVfeedback@hp.com
- HP Business Copy EVA: BCfeedback@hp.com
- HP Continuous Access EVA: CAfeedback@hp.com



---

# 1 Getting started

HP StorageWorks Storage System Scripting Utility (hereafter called the utility) is a command line interface that allows you to configure and control EVA arrays. Use the utility to script and run repetitious and complex configuration tasks. Use HP StorageWorks Command View EVA, the graphical user interface, for simple or initial configuration tasks

This chapter includes information on how to install and use the utility. The following topics are covered:

- Description
- Getting Started
- Security Considerations
- Installing the Storage System Scripting Utility
- Starting the Storage System Scripting Utility
- Understanding paths and naming conventions
- Aliases
- Formatting and Output of Return Data
- Replica Resync with Snapclones
- Creating Snapclones
- Cross Vraid



---

## NOTE:

The CAPTURE CONFIGURATION command is the only way to capture, save, and recreate a storage system's configuration. After you have successfully created a storage system, use the CAPTURE CONFIGURATION command to create scripts that you can use to recreate a storage system in the event of failure or to create an exact replica.

---



---

## NOTE:

Although this guide shows commands spanning more than one line, always enter the command on one line.

---



---

## NOTE:

*OS\_UNIT\_ID and CONSOLE\_LUN\_ID*

This value is used for IBM AIX (set to zero), OpenVMS (required), and Tru64 UNIX (recommended). Other host operating systems ignore the value.

---



---

## NOTE:

With this release of the utility, the SHOW commands have been replaced with the LS commands. The SHOW commands remain available, but are deprecated.

---



---

**NOTE:**

When a controller pair has been password-protected and the password has not been added in HP Command View EVA, the utility commands will display:

Error: API unable to get the lock

---



---

**NOTE:**

In when you are using the utility, you can not move objects between folders.

---



---

**NOTE:**

In rare instances, the utility loses communication with HP StorageWorks Command View EVA and reports that it is down, stopped, or restarting even though it is possible to browse to HP StorageWorks Command View EVA 4.0. To fix this, restart HP StorageWorks Command View EVA.

---



---

**NOTE:**

On the HSV200 controllers only, the fabric port names in the output from the HP EVAPerf CLI's `ps` command are reversed. FP1 should be FP2 and vice versa, for both controllers.

---



---

**NOTE:**

You may experience a condition where the symptom is a `No Object Found error` or a `Neither Success or Failure error`.

To resolve this:

- Wait a few seconds and the problem may resolve itself
  - or
  - Close all of your open browser windows, including the HP StorageWorks Command View EVA windows. The utility does not need to be restarted. If the problem persists after these steps, restart HP StorageWorks Command View EVA v4.0.
- 

## Getting started

To get started using the utility, follow these steps:

1. Select an HP Command View EVA instance (or server) using the `SELECT MANAGER` command. Use the `SELECT MANAGER` command to direct the command prompt to a specified server. All configuration commands run on the selected array and affect the array's configuration. The `<manager_name>` can be a server or an IP address. You can also use `localhost` if you are logged directly on the management server. Unless the storage administrator has changed the password, the default password is "administrator" and the default user is "administrator".  
Syntax (to select the HP Command View EVA instance — or server):

```
SELECT MANAGER <manager_name> USERNAME=<username> PASSWORD=<password>
```

2. View available arrays using the `LS SYSTEM` command Syntax: `LS SYSTEM`
3. Select an array to manage using the `SELECT SYSTEM` command Syntax: `SELECT SYSTEM <system_name>`

# Security considerations

## ProLiant servers and Smartstart 7.2

If you will be running the utility on a ProLiant server, please read “Password security incompatibility” in the HP StorageWorks Command View EVA installation guide

For the latest versions of storage documentation, access the HP storage web site at <http://www.hp.com/country/us/eng/prodserv/storage.html>.

## Installing the utility

This section covers installing the utility.

### Installation

1. For Windows, the utility is installed when you install HP Command View EVA in:

C:\Program Files\Hewlett-Packard\SANworks\Element Manager for StorageWorks HSV\Bridge

If you are using other operating systems, you need to download the software at the following url:

<http://h18006.www1.hp.com/products/storage/software/cmdvieweva/index.html>

2. Unzip the file and copy the binary to the location of your choice.

Once installed, the utility runs as a simple executable. For convenience, you can add the directory containing the utility executable to your path or copy the executable to a directory already in your path. Ensure that the utility executable's attributes are set with the correct security and execution flags appropriate for your environment.

## Starting the utility

The utility starts as a command prompt window, shell, or equivalent. To start utility, enter the following command prompt:

**sssu.exe**



---

**NOTE:**

The **SET OPTIONS** command (see [SET OPTIONS](#)) controls how utility behaves. The options you set with the **SET OPTIONS** command are in effect for the current session only. Each time you start the utility, the default options are reinstated.

---

## Syntax

SSSU <additional arguments>

If the utility is started without arguments, a generic application <NoSystemSelected> prompt is displayed on the terminal and input can be accepted.

If the utility is started with arguments, those commands are echoed to the input terminal and executed, and then the utility exits.

**NOTE:**

You must enclose in double quotes any commands or object names that have embedded blanks (spaces).

## Examples

```
SSSU
```

This example starts the utility without additional arguments and prompts you for commands.

```
SSSU "FILE snapD1.txt"
```

This example starts the utility and then executes the file `snapD1.txt` from the current directory.

```
SSSU "cmd1" "cmd2" "cmd3"
```

This example starts the utility and executes multiple commands.

## Shortcut commands

The utility commands and switches can be shortened to the length that makes them unique. For example, a shortcut for `EMVERSION` could be `EM` (note that using just the letter `E` would not make the command unique since `EXERCISE_DISK` also starts with the letter `E`).

## Understanding paths and naming conventions

This section describes important information about paths and naming conventions in the utility.

**NOTE:**

For detailed descriptions of the following commands, refer to Chapter 2.

The following important root folders let you organize your array:

You cannot create root-level folders. You can, however, create additional folders within the `\Hosts\` and `\Virtual Disks\` folders to organize your array (see [ADD FOLDER](#))

- `"\Hosts\"`
- `"\Disk Groups\"`
- `"\Data Replication\"`
- `"\Hardware\"`
- `"\Virtual Disks\"`

## Virtual disk example

Due to the space in the root name for the Virtual Disks folder, you must always enclose strings that include the name of this folder within double quotes (`"`).

```
ADD VDISK "\Virtual Disks\new_code" SIZE=10
```

This example creates a virtual disk family named `new_code` in the default disk group.

This is a virtual disk family. The actual virtual disk name (the active virtual disk) is `\Virtual Disks\new_code\ACTIVE`.



**NOTE:**

If you create a virtual disk (using ADD VDISK) within a deeper folder structure, you need to include the full path within double quotes.

```
ADD VDISK "\\Virtual Disks\engineering\gene_research" SIZE=2
```

---

## Disk groups example

This example uses an alias to identify a disk group.

```
LS disk_group admin
```

## Aliases

The utility commands recognize an alias, so you do not need to use full names except for LUN names.

## Virtual disk example

If you have `\virtual_disks\my_virdisk\Active`, you can enter **my\_virdisk** for the object. If your object name contains spaces, the double quotes are mandatory.

## Disk group example

If you have `\Disk Groups\Default Disk Group`, you can enter **Default Disk Group** for the object.

**NOTE:**

If you do not specify a full name, the utility checks to see if you are using an alias. However, LUNs need to be specified with a full name. This is because LUNs are not uniquely identified by numbers and several different virtual disks can have a LUN 1 presentation. In this instance, the full path name is required.

For example, if you are specifying a LUN, you need to specify the full LUN name:

```
LS lun \Hosts\MainServer\3
```

---

## SET example

To specify a host without aliases:

```
SET host \Hosts\h1 name=h2
```

To specify a host with aliases:

```
set host h1 name=h2
```

## Add snapshot example

To add a snapshot without aliases:

```
add snapshot mysnap vdisk="\Virtual Disks\mydisk\ACTIVE"
```

To add a snapshot with aliases:

```
add snapshot mysnap2 vdisk=mydisk
```

## Delete virtual disk example

To delete a virtual disk without aliases:

```
delete vdisk "\Virtual Disks\mydisk\ACTIVE"
```

To delete a virtual disk with aliases:

```
delete vdisk mydisk
```

## Formatting and output of returned data

Most return LS command data can be formatted in the optional XML format.

The regular output of the LS command is in the format:

```
XML element : value
```

To specify XML output, add the XML switch. For example, the following command:

```
LS vdisk vd01 XML
```

provides the following XML output:

```
<objectname>\Virtual Disks\vd01\ACTIVE</objectname>
```



---

**NOTE:**

The LS commands provide a flexible format to display system data. The SHOW commands are still available for script compatibility, but they are deprecated (available, but not supported).

---

## Replica resync with snapclones

Replica resync allows you to revert back to a previous point-in-time (snapclone) if a virtual disk becomes corrupted. The process involves changing the corrupted virtual disk into a container (using the SET VDISK CHANGE\_INTO\_CONTAINER command). This will create an empty container with the same settings as the original virtual disk. You can then restore the virtual disk data by creating a snapclone of the previously taken backup snapclone. When the new snapclone completes, it will become an independent virtual disk with the same settings as the previously corrupted virtual disk. Your data will be as current as your most recent backup snapclone.

## Creating snapclones

With this release of the utility, a snapclone of a virtual disks can be preallocated using containers. Containers provide a more efficient way to create snapclones because they can be created before they are needed for a snapclone. Additionally, reserving the disk space ensures the amount of space needed for the snapclone will be immediately available when it is needed. Containers are created using the ADD CONTAINER command.

## Cross Vraid

Cross Vraid for snapshot and snapclone allow customers to change the Vraid type when creating snapshots or snapclones for better disk utilization. Cross Vraid is supported in the following manner:

- Cross Vraid Snapshot (ADD SNAPSHOT) and Snapclones (ADD COPY) within the same disk group.
- Cross Vraid Snapclones (ADD COPY) across disk groups (including standard and FATA disk groups)

Refer to ADD COPY and ADD SNAPSHOT for more information.



---

**NOTE:**

You cannot create a snapshot of a RAID level that is more available than the parent Vdisk. For example, you cannot create a snapshot of Vraid1 if the parent Vdisk is Vraid0. The reason is that if the parent Vdisk is lost, the snapshot will not be more available than the parent Vdisk.

Subsequent snapshots of a virtual disk must be of the same redundancy level. For example, if you create a Vraid5 snapshot of a Vraid1 virtual disk, then all other snapshots of the original virtual disk must also be Vraid5. As an example, you could have seven Vraid5 snapshots of the original Vraid1 virtual disk.

Vraid availability in order, from highest to lowest is: Vraid1, Vraid5, and Vraid0.

The maximum number of snapshots you can create for a single virtual disk is 7.

---



---

## 2 Utility command reference

This chapter defines all the commands and switches available in the utility, including:

- ADD
- CAPTURE
- DELETE
- EMVERSION
- EXERCISE DISK
- EXIT
- FILE
- HELP
- LS
- REDISCOVER
- RESTART
- SELECT
- SET
- SHUTDOWN

In addition, this chapter includes syntax and examples. The commands are presented in alphabetical order.



---

### NOTE:

When running an SSSU script from a file, the entire file argument must be contained in double quotes. For example

```
Sssu "file myscript.txt"
```

In addition, if the file path name contains spaces, then it too must be in double quotes and be designated with the ASCII escape character "\". For example:

```
Sssu "file \"c:\program files\myotherscript.txt\""
```

---



---

### NOTE:

The LS commands should be used in place of the SHOW commands. This version of the utility no longer supports the SHOW commands (they are deprecated). However, previous scripts using the SHOW commands will still work.

---

## Command introduction

You can display a list of switches for each command directly within the interface by entering a ? after the command or option name. For example, ADD ? displays all of the switches available with the ADD command, and ADD SYSTEM ? displays the switches available for the ADD SYSTEM command (see [HELP](#)).

Remember when you issue commands:

- You can use the alias to specify names.
- If a path name contains a space, enclose the entire name in double quotes ("").

Chapter 3 — Configuration Example, provides a simple configuration example for using these commands to create and present a single LUN to a host. It also includes sample output from LS commands.



---

**NOTE:**

Command keywords are not case sensitive, and it is often not necessary to type complete command names. For example, you can type **em** for EMVERSION or **shu** for SHUTDOWN. However, user created object names may be case sensitive if they are intentionally created with uppercase and lowercase characters.

---

## Menu map

A menu map is shown on the next few pages. You can use this as a graphical reference for the utility commands.

Command	Switch	Option
ADD	CONTAINER	DISK_GROUP
		REDUNDANCY
		SIZE
	COPY	CONTAINER
		DISK_GROUP
		NOWAIT_FOR_COMPLETION
		OS_UNIT_ID
		REDUNDANCY
		VDISK
	DISK_GROUP	WAIT_FOR_COMPLETION
		WORLD_WIDE_LUN_NAME
	DR_GROUP	COMMENT
		DEVICE_COUNT
		DISKGROUP_DISKTYPE
		OCCUPANCY_ALARM
	DR_GROUP	SPARE_POLICY
		ACCESSMODE
		COMMENT
		DESTINATION_DISK_GROUP
		DESTINATION_SYSTEM
		DESTINATION_VDISK_NAME
		LINK_DOWN_AUTOSUSPEND
		LOG_DESTINATION_DISK_GROUP
		LOG_SOURCE_DISK_GROUP
		MAX_LOG_SIZE
	FOLDER	TARGETREDUNDANCY
		VDISK
	HOST	WRITEMODE
		COMMENT
	LUN	COMMENT
		IP
		OPERATING_SYSTEM
	SNAPSHOT	WORLD_WIDE_NAME
	SYSTEM	HOST
		VDISK
CAPTURE	CONFIGURATION	ALLOCATION_POLICY
		OS_UNIT_ID
DELETE	VALIDATE	REDUNDANCY
		VDISK
	DISK_GROUP	WORLD_WIDE_NAME
		COMMENT
		CONSOLE_LUN_ID
		DEVICE_COUNT
		DISKGROUP_DISKTYPE
		SPARE_POLICY
	VDISK	COMMENT
		DISK_GROUP
		MIRRORCACHE
		NOPERFERED_PATH
		NOREAD_CACHE
		NOWAIT_FOR_COMPLETION
		NOWRITE_PROTECT
		OS_UNIT_ID
		PREFERRED_PATH
		REDUNDANCY
	WAIT_FOR_COMPLETION	READ_CACHE
		SIZE
		WORLD_WIDE_LUN_NAME
		WRITECACHE
	NOWAIT_FOR_COMPLETION	WRITE_PROTECT

Figure 1 Menu map (part A)

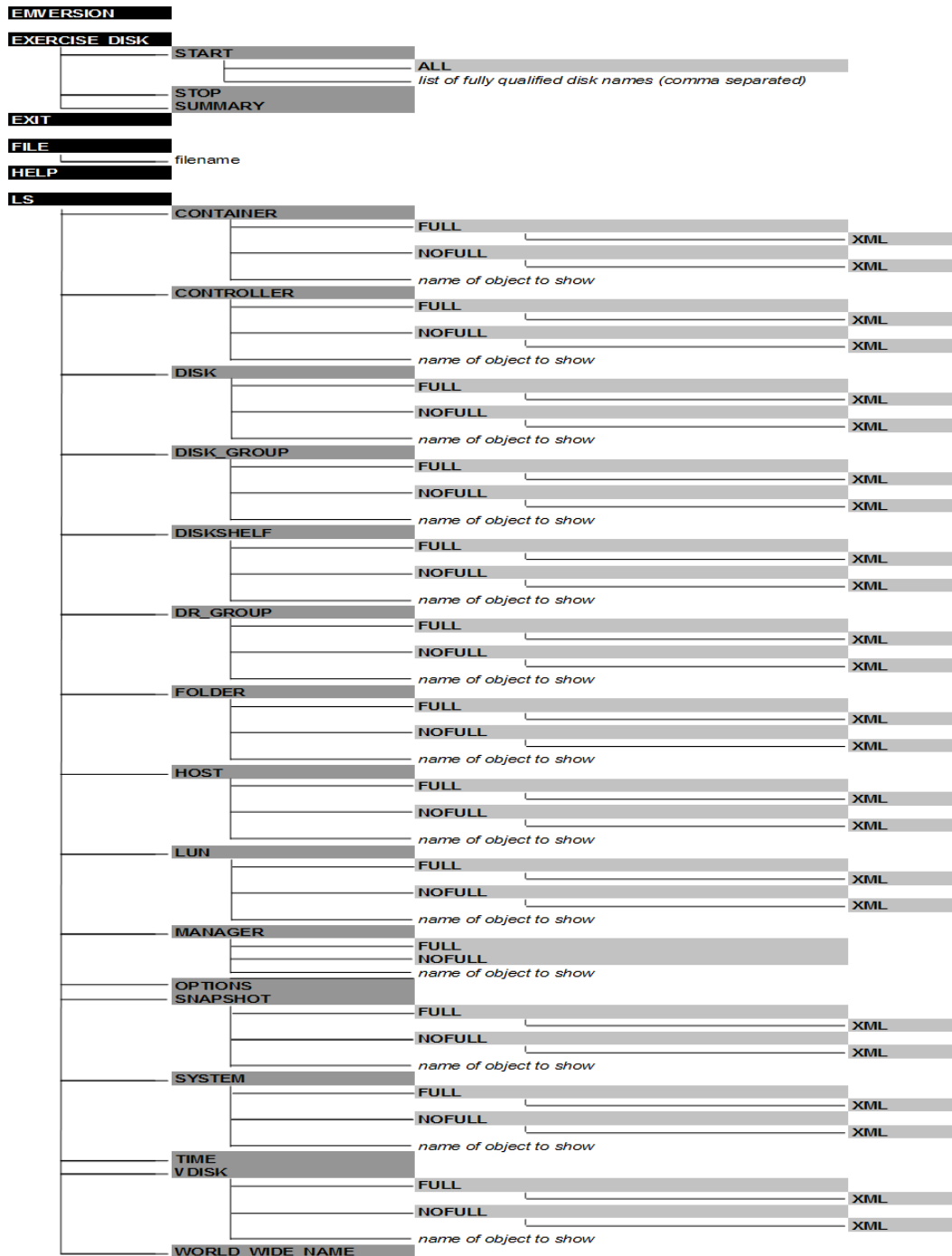
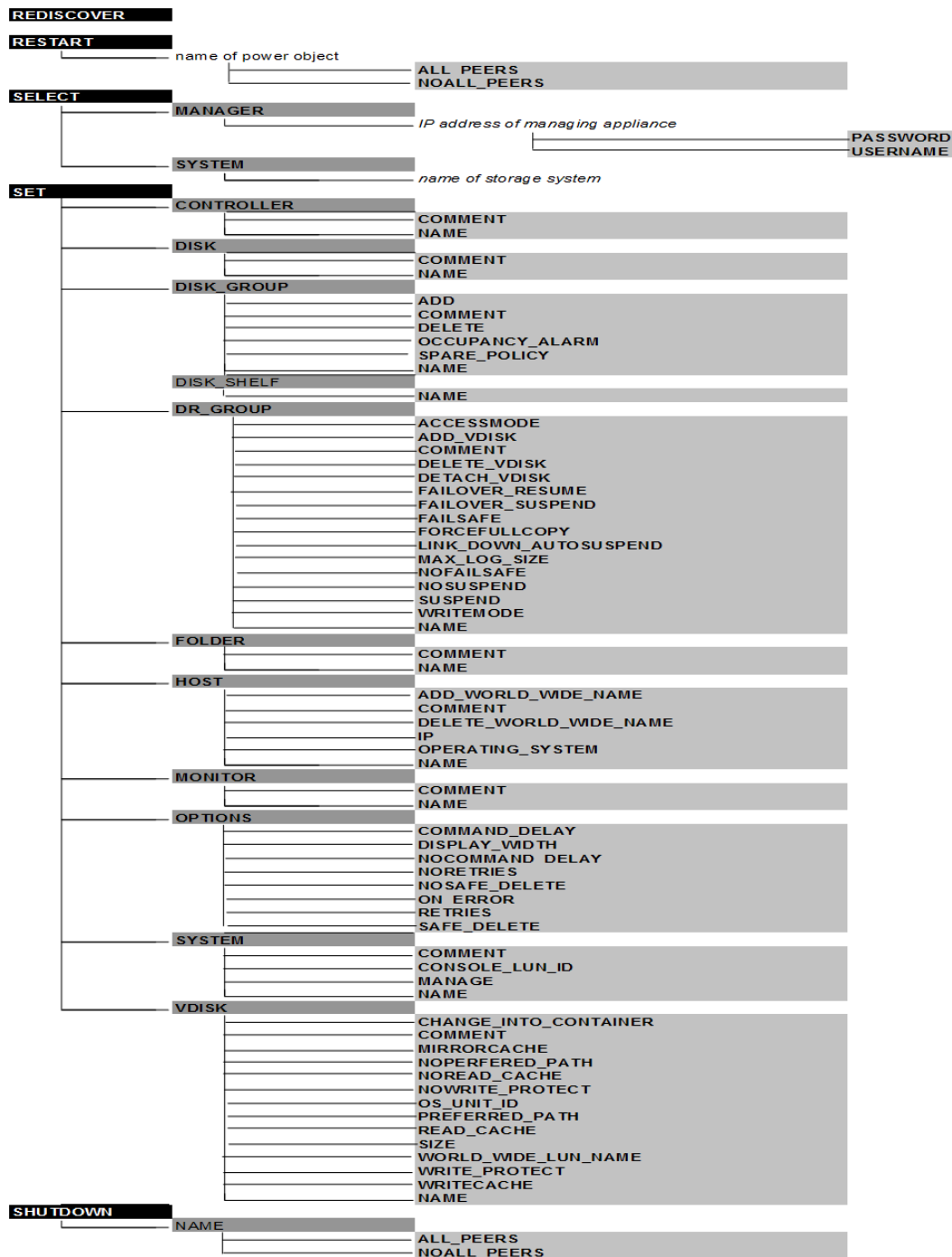


Figure 2 Menu map (part B)





**Figure 3 Menu map (part C)**

## ADD

Use the ADD commands to create new arrays, folders, disk groups, DR Groups, virtual disks, snapshots, snapclones, as well as to add LUNs and hosts within the EVA.

**NOTE:**

Creating DR groups and snapshots of virtual disks is dependent on your licensing. See the documentation that came with your hardware for more information.

## ADD CONTAINER

Use the ADD CONTAINER command to reserve disk space to create copies of virtual disks (snapclones). You can create containers and reserve the space before they are needed for a snapclone. Reserving space allows you to quickly and safely create a copy because the space for the copy is already reserved and available. You must specify the size of the container when you create it (there is no default).

After the copy is created, it becomes another virtual disk. You can reverse the process and turn a virtual disk back into a container using the SET VDISK CHANGE\_INTO\_CONTAINER command.

If the disk group is not specified, it is created in the default disk group. If the default disk group does not exist, an error message will be generated.

**NOTE:**

When you delete a container, the option WAIT\_FOR\_COMPLETION is ignored. After deletion, the container may still appear during an LS command. This is because the container is deleted in the background and will continue to appear until it is fully deleted (this is an asynchronous only operation).

## Syntax

```
ADD CONTAINER <container_name> SIZE=<n>
```

## Switches

The following switches are available with the ADD CONTAINER command.

### DISK\_GROUP=

This is the name of the disk group where you want to create the container.

### REDUNDANCY=

The amount of data protection set when creating the virtual disk. If not specified, the default is Vraid1.

- Vraid0—Provides no data protection. It distributes data among its member disks into stripes and uses all members to process I/O requests. This method has no overhead associated with duplication of information and provides the highest performance.
- Vraid1—Provides the highest level of data protection but uses the most space. It duplicates data written to one disk onto a backup disk. In a multidisk configuration, Vraid1 mirrors each pair of disks to each other. These disk pairs can then be striped to create a virtual disk.
- Vraid5—Provides a moderate level of data protection. This method distributes the parity information among all disk members. If one drive fails, the failed disk can be recreated after it is replaced.

### SIZE=

This is size of the container. Enter decimal value in GB.

## ADD COPY

Use the ADD COPY command to create a copy of the specified virtual disk. A copy is a new virtual disk family. The ADD COPY command is equivalent to creating a snapclone within HP Command View EVA.



### NOTE:

Creating copies of virtual disks depends on your licensing. Attempting to add a copy without the license returns an error message that indicates that you need to enter your licensing information in the HP Command View EVA. You must have a registered license.

## Syntax

```
ADD COPY <copy_name> VDISK= <active member of a vdisk family>
```

## Switches

The following switches are available with the ADD COPY command.

### CONTAINER=

This is the name of the container you would like to use for the virtual disk copy. The container must already exist to use this switch. The container option is optional. If not specified, the add copy command will still function by dynamically allocating the storage before starting the copying process.

### DISK\_GROUP=

This is the name of the disk group where you want to create the virtual disk copy. The disk group must already exist to use this switch. If not specified, the utility uses the same group as the source virtual disk.



### NOTE:

The source virtual disk must be set to `writetocache=writethrough` before a preallocated snapclone (using a container) can be created. See SET VDISK section for more information.

### OS\_UNIT\_ID=

The ID that is presented to the host operating system. If set to zero, no ID is presented to the host.

### REDUNDANCY=

The amount of data protection set when creating the virtual disk. The default is the same as the source virtual disk.

- Vraid0—Provides no data protection. It distributes data among its member disks into stripes and uses all members to process I/O requests. This method has no overhead associated with duplication of information and provides the highest performance.
- Vraid1—Provides the highest level of data protection but uses the most space. It duplicates data written to one disk onto a backup disk. In a multidisk configuration, Vraid1 mirrors each pair of disks to each other. These disk pairs can then be striped to create a virtual disk.
- Vraid5—Provides a moderate level of data protection. This method distributes the parity information among all disk members. If one drive fails, the failed disk can be recreated after it is replaced.

VDISK=

This is the virtual disk name to be copied (active member of a virtual disk family).

WORLD\_WIDE\_LUN\_NAME=

Sets the World Wide LUN Name.

WAIT\_FOR\_COMPLETION

Waits until the command's operation finishes before displaying the next utility prompt (and returning control) or before running another script line. For large (greater than 1 TB) virtual disks, using this option can result in a long period before the prompt appears.

NOWAIT\_FOR\_COMPLETION

Returns control as soon as possible. Some operations are invalid until the initial operation finishes in the background.

## Required Switch

The following switch is required with the ADD COPY command:

VDISK=

### Examples

```
ADD COPY wednesday_nite VDISK="\Virtual Disks\payroll\ACTIVE"
```

This example copies the virtual disk payroll to a copy named wednesday\_nite.

```
ADD COPY save_reports VDISK="\Virtual Disks\daily_biz\ACTIVE"
```

```
DISK_GROUP="\Disk Groups\small_disks"
```

This example creates a copy of daily\_biz named save\_reports within the specified disk group.



---

### NOTE:

When you use the ADD COPY command, you are actually creating a virtual disk. If you want to delete the virtual disk, you must use the DELETE VDISK command.

---

## ADD DR\_GROUP

Use the ADD DR\_GROUP command to create a data replication group containing the specified source virtual disk. This source virtual disk is replicated on the specified destination array.

### Syntax

```
ADD DR_GROUP <dr_group_name> VDISK=<vdisk> DESTINATION_SYSTEM=<system>
```

### Switches

The following switches are available with the ADD DR\_GROUP.

ACCESSMODE=

The access rights for a connected host. Possible values are: presentonly, readonly, and disable. The default is disable, and this is the safest setting.



---

**NOTE:**

Setting `ACCESSMODE` to `readonly` or `presentonly` could seriously impede host OS operation.

---



---

**NOTE:**

The default *disable* is displayed in the GUI as *none*.

---

#### `COMMENT=`

Used to attach comments to the DR Group being created. The comment string must be enclosed in quotes. The maximum number of characters for this optional switch is 64.

#### `DESTINATION_SYSTEM=`

The system on which the destination virtual disks are created.

#### `DESTINATION_DISK_GROUP=`

The disk group name in which the virtual disk on the destination array is created.

#### `DESTINATION_VDISK_NAME=`

The name of the virtual disk that is created on the destination array. The default name is the same as that of the source.

#### `LINK_DOWN_AUTOSUSPEND=`

When enabled, the DR replication will be automatically suspended if the link between the arrays goes down. It will stay suspended even if the link returns. The available values are: **Enable** and **Disable**.

#### `LOG_DESTINATION_DISK_GROUP=`

This is the disk group for the DR log disk on the destination array.

#### `LOG_SOURCE_DISK_GROUP=`

This is the disk group for the DR log disk on the source.

#### `MAX_LOG_SIZE=`

This is the maximum size for the DR log disk. The value is the number of blocks in 512 bytes. If left blank or set to 0 (zero), the controller firmware will calculate the best log size for the space available. Otherwise, you can specify a size, which must be between 136MB and 2TB entered in blocks: 278528 through 4294967296.



---

**NOTE:**

If you specify a size that is larger the available disk space, the command will complete successfully, and the DR group will be created (even though the space is not available). If additional disks are added to the configuration, the controller will allocate more data for the DR log.

---

#### `TARGETREDUNDANCY=`

This is the redundancy of the remote virtual disk that will be created on the destination array.

### VDISK=

The name of a source system virtual disk to be placed within the DR Group. A destination virtual disk is automatically created on the destination array.

### WRITEMODE=

Defines the I/O interaction between the destination and source. Possible values are synchronous and asynchronous. The default is synchronous.

### Example

```
ADD DR_GROUP Transactions DESTINATION_SYSTEM=remote_scell  
    VDISK="/Virtual Disks/Hawaii/ACTIVE"
```

This example creates a DR Group named `Transactions` that contains the source virtual disk `\Virtual Disks\Hawaii\ACTIVE`. This DR Group will be connected to the destination array `remote_scell`, on which the destination virtual disk `\Virtual Disks\Hawaii\ACTIVE` (same as the source by default) will be created and added to the destination DR Group.

## Required Switches

The following switches are required for the `ADD DR_GROUP` command.

- `DESTINATION_SYSTEM =`
- `VDISK =`

## ADD DISK\_GROUP

Use the `ADD DISK_GROUP` command to add disk groups to the array.

### Syntax

```
ADD DISK_GROUP <group_name> DEVICE COUNT= <a number>
```

### Switches

The following switches are available with the `ADD DISK_GROUP` command.

### COMMENT=

This is an optional switch that associates a user-defined comment with the new system. A maximum of 128 characters is allowed, and the string must be enclosed in double quotes.

### DEVICE\_COUNT=

The number of physical disks to use for this group. The limit is the number of available disks in the array. The default and minimum value is 8.

### DISKGROUP\_DISKTYPE=

This parameter determines the types of disks to be considered for creating the disk group. The default value is `ONLINE`.

- `ONLINE`: Only online fibre channel disks are considered for creating the default disk group.
- `NEAR-ONLINE`: Only near-online fibre channel disks are considered for creating the default disk group.

## OCCUPANCY\_ALARM=

The point when a defined percentage of space is used. When this point is reached, an event log is generated and sent to the management server (optionally the host system) informing the administrator that the group is becoming full. Do not use the percentage sign (%) after the number. The default is 95%.

## SPARE\_POLICY=

This parameter determines the amount, if any, of storage space set aside for use in the event that disks fail. The default is single.

- NONE—Reserves no space within a disk group to allow for data reconstruction in case of failure of disk drives



### NOTE:

Using a spare policy of NONE can cause data loss and is not recommended.

- SINGLE—Reserves space within a disk group to allow for data reconstruction for failure of a single disk drive
- DOUBLE—Reserves space within a disk group to allow for data reconstruction for failure of two disk drives



### NOTE:

The space set aside is not in numbers of physical disks. It is the equivalent amount of storage space spread across all disks.

## Example

```
ADD DISK_GROUP "\Disk Groups\human_resources" DEVICE_COUNT=12  
  
SPARE_POLICY=SINGLE OCCUPANCY_ALARM=75
```

This example creates a new disk group, `human_resources`. It has 12 physical disks with the equivalent of one disk set aside as a spare and sends an event log to the array when 75% of capacity is reached.

## ADD FOLDER

Use the ADD FOLDER command to create a new folder within the specified folder to aid in organizing your array. You can create folders only under the `Virtual Disks` and `Hosts` root folders. You cannot create root folders.

For example, if you have a controller that is serving Human Resources (HR) and Engineering, you could create four folders—two to separate the virtual disks and two to separate the hosts:

- `\Virtual Disks\Engineering and \Virtual Disks\HR`
- `\Hosts\Engineering and \Hosts\HR`

Creating these folders allows you to put engineering virtual disks and hosts in the Engineering folders and Human Resources storage\hosts in the HR folders. This makes it easier to keep track of the components within your array. If you want to nest folders inside folders, you must add one folder at a time.

## Syntax

```
ADD FOLDER <folder_name>
```

## Switch

Only the COMMENT switch is available with the ADD FOLDER command.

### COMMENT=

This is an optional switch that associates a user-defined comment with the new array. A maximum of 128 characters is allowed, and the string must be enclosed in double quotes.

### Examples

```
ADD FOLDER \Hosts\human_resources
```

This example creates a new folder `human_resources` within the root folder `Hosts`.

```
ADD FOLDER "\\Virtual Disks\colorado\colorado springs\engineering"
```

In this example, the folder structure must already have been in place before creating the new folder `engineering`.

## ADD HOST

Use the ADD HOST command to add a host and its World Wide Name (WWN) to the list of hosts that can connect to virtual disks within the current array.

The ADD HOST command adds one Fibre Channel Adapter (FCA) only. The [SET HOST](#) command adds each subsequent FCA.

## Syntax

```
ADD HOST <host_name> WORLD_WIDE_NAME= <world_wide_name>
```

## Switches

The following switches are available with the ADD HOST command.

### IP=

This is the network IP address. If the IP address is not specified, the name of the host will be used as a DNS name.

### COMMENT=

This is an optional switch that associates a user-defined comment with the new system. A maximum of 128 characters is allowed, and the string must be enclosed in double quotes.

### OPERATING\_SYSTEM=

The type of operating system for the specified host. For the default operating system, refer to Agent Options - User Interface Options in the user interface. Choose one of the following operating systems (they are spelled here as the utility expects):



- CUSTOM

The correct syntax is

```
OPERATING_SYSTEM=CUSTOM=<16 hex characters>
```

- HPUX
- IBMAIX
- OPEN\_VMS
- SOLARIS
- TRU64
- LINUX
- UNKNOWN
- WINDOWS

#### WORLD\_WIDE\_NAME=

The World Wide Name of the Fibre Channel adapter.

#### Required switch

The WORLD\_WIDE\_NAME switch must be set to ensure that HP Command View EVA recognizes the host added with the ADD HOST command.

#### Example

```
ADD HOST \Hosts\development WORLD_WIDE_NAME=5000-1fe1-ff00-0000
```

This example adds a host development with an adapter at a WWN of 5000-1fe1-ff00-0000.

## ADD LUN

The ADD LUN command makes previously created virtual disks available to a host.

#### Syntax

```
ADD LUN <LUN_number> VDISK=<storage_name> HOST=<host_name>
```

#### Switches

The following switches are available with the ADD LUN command.

#### HOST=

The name of the host to which the LUN is presented.

#### VDISK=

The name of the virtual disk that is presented to the host.

#### Required switches

Set the following switches so that HP Command View EVA recognizes the LUN you add with the ADD LUN command.

- HOST
- VDISK

## Examples

```
ADD LUN 12 VDISK="\Virtual Disks\act\payroll\ACTIVE" HOST=\Hosts\sanfran
```

This example adds LUN 12, which presents the ACTIVE virtual disk for the family payroll to the host called sanfran.

```
ADD LUN 175 VDISK=USER_DISK HOST= corporate
```

This example uses aliases and adds LUN 175, which presents the ACTIVE virtual disk for the family user\_disk to the host called corporate.

## ADD SNAPSHOT

The ADD SNAPSHOT command creates a picture (not a full copy) of a virtual disk family's active virtual disk.



### NOTE:

Creating snapshots of virtual disks depends on your licensing. Attempting to add a snapshot without the license returns an error message. This message indicates that you need to enter your licensing information in the HP Command View EVA. You must have a registered license.

## Syntax

```
ADD SNAPSHOT <snapshot_name> VDISK=<vdisk_to_snapshot>
```

## Required switch

The following switch is required with the ADD SNAPSHOT command:

**VDISK=**

### Examples

```
ADD SNAPSHOT payroll_backup VDISK="\Virtual Disks\payroll\ACTIVE"
```

```
ALLOCATION_POLICY=DEMAND
```

This example creates a snapshot payroll\_backup from the ACTIVE member of virtual disk family payroll that uses capacity only as needed.

```
ADD SNAPSHOT wed_nite_biz VDISK=daily_biz ALLOCATION_POLICY=FULLY
```

This example creates a snapshot wed\_night\_biz from the ACTIVE member of the family daily\_biz while reserving all capacity necessary to create the snapshot.



### NOTE:

To delete snapshots created by ADD SNAPSHOT, use the DELETE VDISK command.

## Switches

The following switches are available with the ADD SNAPSHOT command.

**ALLOCATION\_POLICY=**

Indicates how the space for the snapshot is allocated.

- DEMAND—Storage is allocated for the snapshot only when required. As the snapshot and the original storage's information diverges, the space allocated for the snapshot increases.
- FULLY—All storage required to fully contain a snapshot is allocated when the snapshot is created. This is the conservative method to create a snapshot, because adequate space is guaranteed for a full snapshot. This is the default.

#### OS\_UNIT\_ID=

The ID that is presented to the host operating system. If set to zero, no ID is presented to the host.

#### REDUNDANCY=

The amount of data protection set when creating the virtual disk. If not specified, the default is the setting of the source virtual disk.

- Vraid0—Provides no data protection. It distributes data among its member disks into stripes and uses all members to process I/O requests. This method has no overhead associated with duplication of information and provides the highest performance.
- Vraid1—Provides the highest level of data protection but uses the most space. It duplicates data written to one disk onto a backup disk. In a multidisk configuration, Vraid1 mirrors each pair of disks to each other. These disk pairs can then be striped to create a virtual disk.
- Vraid5—Provides a moderate level of data protection. This method distributes the parity information among all disk members. If one drive fails, the failed disk can be recreated after it is replaced.

#### VDISK=

The source virtual disk for this snapshot.

#### WORLD\_WIDE\_LUN\_NAME=

Sets the World Wide LUN Name.

## ADD SYSTEM

Use the ADD SYSTEM command to initialize an uninitialized array and to create a default disk group. You must select an uninitialized array before issuing the ADD SYSTEM command. If the system is already initialized, the command is rejected.

Using the LS SYSTEM command (see [LS SYSTEM](#)) to display the names of the available uninitialized arrays, the uninitialized array displays as Uninitialized Storage System [WWN of the storage cell].

After initialization is complete, the utility changes its default prompt back to NoSystemSelected.

Because the array's name changed from the uninitialized string to the name given with the ADD SYSTEM command, you must reselect the array. Issue the SELECT SYSTEM command using the new name of the array.



#### NOTE:

Appendix A includes a configuration example that explains the sequence of commands you can use to initialize an uninitialized array.

## Syntax

```
ADD SYSTEM <system_name> DEVICE_COUNT=
```

## Switches

The following switches are available with the ADD SYSTEM command.

### COMMENT=

This is an optional switch that associates a user-defined comment with the new array. A maximum of 128 characters is allowed, and the string must be enclosed in double quotes.

### CONSOLE\_LUN\_ID=

The LUN used for console communication after array creation. If set to zero, no console LUN is presented to the host.

### DEVICE\_COUNT=

The number of physical disks to use for the default disk group of this array. The limit is the number of available disks in the array. The default and minimum value is 8.

### DISKGROUP\_DISKTYPE=

This parameter determines the types of disks to be considered for creating the disk group. The default value is ONLINE.

- ONLINE: Only online fibre channel disks are considered for creating the default disk group.
- NEARONLINE: Only near-online Fibre channel disks are considered for creating the default disk group.

### SPARE\_POLICY=

This parameter determines the amount, if any, of storage space set aside for safety in the event that disks fail.

The default is SINGLE.

- NONE—Reserves no space within a disk group to allow for data reconstruction for failure of disk drives.
- SINGLE—Reserves space within a disk group to allow for data reconstruction for failure of a single disk drive.
- DOUBLE—Reserves space within a disk group to allow for data reconstruction for failure of two disk drives.



---

#### NOTE:

The space set aside with spare policy is not in numbers of physical disks. It is the equivalent amount of storage space spread across all disks.

---

### Example

```
ADD SYSTEM payroll DEVICE_COUNT=12 SPARE_POLICY=SINGLE
```

This example creates an initialized array `payroll` with 12 physical disks and a spare policy of SINGLE.

## ADD VDISK

Use the ADD VDISK command to create a virtual disk with a specified name and parameters.



---

**NOTE:**

This command actually creates a virtual disk family and the ACTIVE member. After you create the family, an initial virtual disk, "ACTIVE," is placed as the only virtual disk in the family. This is referred to as the active virtual disk.

---

## Syntax

```
ADD VDISK <storage_name> SIZE=<n>
```

The *n* is the virtual disk size in whole GBs.

## Switches

The following switches are available with the ADD VDISK command.

### COMMENT=

This is an optional switch that associates a user-defined comment with the new virtual disk. A maximum of 128 characters is allowed and the string must be enclosed in double quotes.

### DISK\_GROUP=

The disk group name for where you want to create the virtual disk. The disk group must already exist to use this switch. If not specified, the default disk group is used.

### NO\_WAIT\_FOR\_COMPLETION

This option returns control of the array while the virtual disk is being created and before it is complete.

### OS\_UNIT\_ID=

The ID that is presented to the host operating system. If set to zero, no ID is presented to the host.

### PREFERRED\_PATH=

The preferred controller path that is specified to handle all I/O for the virtual disks. If a controller fails, the path always reverts to the working controller.

- **PATH\_A\_BOTH**—Controller path A fails over to controller B. When controller A restarts, the virtual disks fail back to controller A. This is failover/failback mode.
- **PATH\_A\_FAILOVER**—Controller path A fails over to controller B. When controller A restarts, the virtual disks do not fail back to controller A. This is failover-only mode.
- **PATH\_B\_BOTH**—Controller path B fails over to controller A. When controller B restarts, the virtual disks fail back to controller B. This is failover/failback mode.
- **PATH\_B\_FAILOVER**—Controller path B fails over to controller A. When controller B restarts, the virtual disks do not fail back over to controller B. This is failover-only mode.

### NOPREFERRED\_PATH

Allows the I/O to be handled by either controller.

### READ\_CACHE

Reads are satisfied from the controller's cache.

## NOREAD\_CACHE

Reads are always satisfied from the physical disks, not the controller's cache.

## REDUNDANCY=

The amount of data protection set when creating the virtual disk. If not specified, the default is changed to Vraid1.

- Vraid0—Provides no data protection. It distributes data among its member disks into stripes and uses all members to process I/O requests. This method has no overhead associated with duplication of information and provides the highest performance.
- Vraid1—Provides the highest level of data protection but uses the most space. It duplicates data written to one disk onto a backup disk. In a multidisk configuration, Vraid1 mirrors each pair of disks to each other. These disk pairs can then be striped to create a virtual disk.
- Vraid5—Provides a moderate level of data protection. This method distributes the parity information among all disk members. If one drive fails, the failed disk can be recreated after it is replaced.

## WORLD\_WIDE\_LUN\_NAME=

Sets the World Wide LUN Name.



---

### NOTE:

This switch is commonly used to allow a host to point to a new version of the virtual disk by giving the new virtual disk the same WWN as the old virtual disk.

---

## WRITE\_PROTECT

Does not allow writing to the virtual disk for all presented LUNs/host.

## NOWRITE\_PROTECT

Turns off write protection and allows writing to the virtual disk for all presented LUNs/host.

## WAIT\_FOR\_COMPLETION

Default. Waits until the command's operation completes before displaying the next utility prompt (and returning control) or before running another script line. For large (greater than 1 TB) virtual disks, using this option can result in a long period before the prompt displays.

## WRITECACHE=

This is the cache parameter for the write operation. There are two values: WRITETHROUGH and WRITEBACK. With WRITETHROUGH the operation completes when the write to disk completes. With WRITEBACK, the operation completes when the data is written to cache.



---

### NOTE:

If you are creating a preallocated snapclone (using a container), you need to change the Write Cache Policies from WRITEBACK to WRITETHROUGH for the source virtual disk. The reason the cache policy has to be changed to WRITETHROUGH is to flush any cache memory to the virtual disk before it is cloned. The policy of the source virtual disk will revert back to the original setting when you complete the creation of the snapclone.

---

## MIRRORCACHE=

This sets the controller's mirror cache. The values are MIRRORRED (cache is mirrored between both controllers) and NOTMIRRORRED (cache is not mirrored).

## Required switch

The SIZE switch is required with the ADD VDISK command.

## Switch defaults

The following defaults are used for the ADD VDISK command when nothing is specified:

- MIRRORCACHE=Mirrored
- WRITECACHE=WriteBack
- OS\_UNIT\_ID= defaults to zero
- PREFERRED\_PATH= defaults to NOPREFERRED\_PATH
- REDUNDANCY= defaults to Vraid1
- DISK\_GROUP= \Disk Groups\Default Disk Group
- READ\_CACHE
- NOWRITE\_PROTECT



### NOTE:

If the Default Disk Group has been renamed, not specifying a disk group will result in error.

## Examples

```
ADD VDISK "\Virtual Disk\scratch" SIZE=10 REDUNDANCY=VRAID5  
READ_CACHE
```

This example creates a 10-GB virtual disk scratch within the default disk group using read cache and a Vraid5 redundancy level.

```
ADD VDISK "\Virtual Disks\engineering\gene_research" SIZE=2  
DISK_GROUP="\Disk Groups\small_disks"
```

This example creates a 2-GB disk at Vraid1 redundancy level in the small\_disksgroup disk group.



### NOTE:

Remember that this name is the <family\_name> and that "\Virtual Disks\ <family\_name> \ACTIVE" is the name of the ACTIVE virtual disk. You must use this full path (enclosed in double quotes) when referring to it with any other commands, or you can use aliases.

# CAPTURE (CONFIGURATION)

## Syntax

```
CAPTURE CONFIGURATION <file_name>
```

CAPTURE CONFIGURATION queries the currently selected array and creates from one to five utility scripts. Not all arrays require all five scripts. For example, step1A is always generated and may be the only file that is required. You use these scripts to recreate the original configuration, and in some cases, to assist in the recovery of a site.

The scripts output to the console unless you specify a file name. The CAPTURE CONFIGURATION command generates five files. Given a file name, `_StepXX` is appended after the file name and before the extension. XX is the restore step name, which is 1A, 1B, 1C, 2, or 3. For example, specifying `CAPTURE CONFIGURATION newyear.txt` causes the utility to create the files `newyear_Step1A.txt`, `newyear_Step1B.txt`, `newyear_Step1C.txt`, `newyear_Step2.txt`, and `newyear_Step3.txt`.

The utility checks to see if any files by the names to be generated exist. If the file names exist, and you are entering the command manually, the utility will prompt you to replace existing files with the new files. If you are using a script, existing files will be replaced with the new files.

When creating files that contain the scripts, activity dots display on the console. This command may take a long time to complete depending on the size of the configuration.



---

**NOTE:**

Do not reconfigure the selected array while this command is executing.

---

## Step 1 scripts

The division of step 1 into three smaller steps facilitates data replication (DR) recovery.

### Step 1A script

Creates the array, disk groups, hosts, and virtual disks that are not used for data replication and LUNs for the created disks. Step1A creates a basic system that does not include DR virtual disks or DR Groups. CAPTURE CONFIG always creates this step.

### Step 1B script

Creates all source virtual disks used in DR Groups on this controller.

### Step 1C script

Presents all source virtual disks (creates LUNS) that are used for DR Groups to their hosts. This step exists to assist in recovering from a DR failure in which a source site was lost. LUNs can be presented in their original configuration by running the correct step 1C script.

## Step 2 script

Recreates all DR-specific configuration information for which this array is the source. This involves the configuration's source DR\_GROUPS and their members only. Presentations of remote virtual disks are not restored by this command (see step 3).

This step provides flexibility when a site is completely lost. Both source and destination have separate CAPTURE CONFIGURATION scripts, so you must run step 1A, step 1B, and step 1C on the source array and then run step 1A, step 1B, and step 1C on the destination arrays before step 2 can be run.



---

**NOTE:**

If no destination virtual disk is presented to a host, a blank script will be generated for script 3.

---



## Step 3 script

Presents remote virtual disks (creates LUNS) used for DR Groups to their hosts.

You must run step 2 on both source and destination arrays before step 3 can be run.

## Restoring configurations

To restore system configuration from captured scripts, you must run the following scripts in the following order on each array that is part of the DR configuration.

Be aware that a specific captured configuration may not be recreated on some arrays. For example, if a captured configuration demands 120 disks and the target array contains only 80, the captured configuration will not be successful.

1. Select a manager to specify which HP Command View EVA will perform the script commands.
2. Select the array Uninitialized Storage System (WWN) to specify the uninitialized storage system on which you want to recreate the captured configuration.
3. Run scripts 1A, 1B, and 1C successfully (and separately) on both source and destination arrays in the DR configuration.
4. Run the step 2 script successfully (and separately) on both source and destination arrays in the DR configuration.
5. Run the step 3 script successfully (and separately) on both source and destination arrays in the DR configuration.

Once these scripts have been run against all arrays, the DR configuration is restored to its original state.

## Example

```
CAPTURE CONFIGURATION c:\sales.txt
```

This example creates the files sales\_Step1A.txt, sales\_Step1B.txt, sales\_Step1C.txt, sales\_Step2.txt, and sales\_Step3.txt.



---

**NOTE:**

Refer to the FILE command for further information about restoring configurations.

---

## CAPTURE (VALIDATE)

### Syntax

```
CAPTURE VALIDATE <file_name>
```

CAPTURE VALIDATE checks the specified script file for modification by calculating its checksum and comparing the result against the checksum saved in the file. A script may have been modified (failed the CAPTURE VALIDATE command) but may still work. The utility runs a script even if its checksum shows file modification.

### CAPTURE VALIDATE example

This example determines whether the file name was modified.

## User functionality for CAPTURE commands

CAPTURE VALIDATE is a switch that determines if a captured script text file has been edited and modified after it was captured. This allows the user to determine whether the script is a perfect capture of the existing array or if someone has modified it after it was created.



---

**NOTE:**

The specified script is not executed by this command; the script is checked only to see whether it is original or not.

---

When a script is captured by the CAPTURE commands, a header is written at the front of the text file and a checksum is written at the end. If CAPTURE VALIDATE does not see a correct header, it reports that the script specified is not a CAPTURE-generated script. If the checksum does not match, this command reports that the file has been modified.

If and only if a correct header is found and the checksum matches, this command reports that the CAPTURE script specified is unmodified.

## DELETE

Use the DELETE commands to remove arrays, folders, disk groups, DR Groups, virtual disks, hosts, systems, and LUNs from the array. You can control the behavior of all DELETE commands with the SET OPTIONS command (see “[SET OPTIONS](#)” for details on how to use the SAFE\_DELETE and NOSAFE\_DELETE switches).

### DELETE DR\_GROUP

Deletes the DR Group on the source and destination array. The SET OPTION NOSAFE\_DELETE does not override this requirement. You can delete a DR Group on the source array at any time.

#### Syntax

```
DELETE DR_GROUP <dr_group_name>
```

#### Example

```
DELETE DR_GROUP "\\Data Replication\payroll"
```

This example deletes the payroll DR Group.

### DELETE DISK\_GROUP

Use the DELETE DISK\_GROUP command to delete a disk group from the configuration. The command is rejected if any virtual disks are present in the disk group.

#### Syntax

```
DELETE DISK_GROUP <group_name>
```

#### Example

```
DELETE DISK_GROUP "\\Disk Groups\human_resources"
```

This example deletes a disk group named human\_resources.

## DELETE FOLDER

Use the DELETE FOLDER command to delete the specified empty folder. The DELETE\_FOLDER command fails if the specified folder is not empty. To delete multiple layers of folders, you must do so sequentially (you cannot delete a root level folder).

### Syntax

```
DELETE FOLDER <folder_name>
```

### Examples

```
DELETE FOLDER \Hosts\human_resources
```

This example deletes a folder named `human_resources` within the Default Hosts folder.

```
DELETE FOLDER \Hosts\colorado\colorado_springs\engineering
```

This example deletes the folder named `engineering` nested in the path `\Hosts\colorado\colorado_springs`.

## DELETE HOST

The DELETE HOST command deletes a host from the array.

### Syntax

```
DELETE HOST <host_name>
```

### Example

```
DELETE HOST \Hosts\development
```

This example deletes the host named `development`.

## DELETE LUN

Use the DELETE LUN command to remove access to a virtual disk through the LUN from a host.

### Syntax

```
DELETE LUN <LUN_name>
```

### Example

```
DELETE LUN "\Hosts\accounting_department\12"
```

This example removes access to LUN 12 from the host `accounting_department`.



---

**NOTE:**

Aliases will not work for LUN names. When you specify a LUN, you need to use the full names.

---

## DELETE SYSTEM

Use the DELETE SYSTEM command to permanently remove the array. The array will no longer be accessible, and all data will be lost.



---

**CAUTION:**

The DELETE SYSTEM command removes the entire array configuration. All customer data on the entire array is lost when you run this command. In addition, all information about LUNs and hosts on this array is lost.

The DELETE SYSTEM command will be rejected if virtual disks exist. Refer to the SET command for the NOSAFE\_DELETE behavior.

---

## Syntax

```
DELETE SYSTEM <system_name>
```

## Example

```
DELETE SYSTEM payroll
```

This example deletes the array named `payroll`.

## DELETE VDISK

Use the DELETE VDISK command to permanently remove the specified virtual disk. The virtual disk is no longer accessible, and all data is lost.



---

**CAUTION:**

The DELETE VDISK command removes the entire virtual disk from the array. All customer data on the specified storage is destroyed when you run this command. In addition, all information about the LUNs presented from this virtual disk is lost as well.

The DELETE VDISK command will be rejected if the virtual disk is presented. See SET OPTIONS — NO\_SAFE\_DELETE for behavior.

---



---

**NOTE:**

When you delete a container, the option WAIT\_FOR\_COMPLETION is ignored. After deletion, the container may still appear during an LS command. This is because the container is deleted in the background and will continue to appear until it is fully deleted (this is an asynchronous only operation).

---

## Syntax

```
DELETE VDISK <storage_name>
```

## Switches

The following switches are available with the DELETE VDISK command.

### WAIT\_FOR\_COMPLETION

Waits until the command's operation completes before displaying the next utility prompt (and returning control) or before running another script line. For large (greater than 1 TB) virtual disks, using this option can result in a long period before the prompt displays.

### NOWAIT\_FOR\_COMPLETION

Default. Returns control as soon as possible. Some operations are invalid until the initial operation completes in the background.

## Examples

```
DELETE VDISK "\\Virtual Disks\d12\ACTIVE" WAIT_FOR_COMPLETION
```

This example deletes ACTIVE and the virtual disk family d12. WAIT\_FOR\_COMPLETION prevents the utility prompt from displaying until the virtual disk deletion is completed.

```
DELETE VDISK "\\Virtual Disks\daily_biz\wed_nite_biz"
```

This example deletes the snapshot wed\_nite\_biz and does not wait for completion.



### NOTE:

You cannot delete ACTIVE if a snapshot of the virtual disk exists. You must delete the snapshot and then delete ACTIVE.

## EMVERSION

The EMVERSION command displays the element manager version and build number.

### Syntax

```
NoSystemSelected: EMVERSION
```



### NOTE:

You will need to select a manager using the SELECT MANAGER command before using the EMVERSION command, but you do not need to select an array.

### Switches

No additional keywords, parameters, or switches are available with the EMVERSION command.

### Example

```
EMVERSION

Element manager information

Version:  4.00

Build:   30
```

## EXERCISE\_DISK

This command allows users to test hard disks in the array for defects.

### Syntax

```
EXERCISE_DISK
```

### Switches

The following switches are available with the EXERCISE\_DISK command.

## START

Begins testing, or exercising, the disks. The options are:

- All  
Tests all disks in a selected array.
- List of fully qualified disk names to exercise, comma separated and enclosed in quotes.  
Tests specific disks as specified in the command.

## STOP

Stops the testing.

## SUMMARY

Provides a summary report of the testing. This can be used to show status while testing is in progress and also when testing is complete.

The following is a sample output:

```
ssl> exer start "\Disk Groups\Ungrouped Disks\Disk 002"
```

```
ssl> exer sum
```

```
DILX Summary Report for Storage Cell ssl
```

```
-----
```

```
DILX process status:  DILX Testing In Progress
```

```
Number of devices tested:  1
```

```
Disk 002
```

```
-----
```

```
Device handle:
```

```
noid:  0x808
```

```
id_type:  0x7
```

```
id_len:  0x10
```

```
id_value[0]:  0x20000004
```

```
id_value[1]:  0xcf792125
```

```
id_value[2]:  0
```

```
id_value[3]:  0
```

```
Device status:  Testing in progress.
```

```
Total blocks transferred:  12098
```

```
Total reads issued:  329
```

```
Total writes issued:  77
```

```
1st bad sense key:  0
```

```
2nd bad sense key:  0
```

```
1st error flag:  0
```

```
2nd error flag: 0
Soft error count: 0
Hard error count: 0
1st bad ASC: 0
2nd bad ASC: 0
1st bad ASQ: 0
2nd bad ASQ: 0
ssl> exer stop
```

## Examples

```
exercise_disk start "\Disk Groups\Default Disk Group\Disk 002,
\Disk Groups\Default Disk Group\Disk 004"
```

This example exercises Disk 002 and Disk 004.

```
exercise_disk start all
```

This example exercises all disks.

## EXIT

The EXIT command terminates the utility session.

If the utility is accepting input from the terminal or the command line, EXIT causes the program to terminate.

If the utility is processing a script from a file while encountering a FILE command, the behavior of the EXIT command depends on how the FILE command was issued:

- If the FILE command was issued from the command line, an EXIT command in the file causes the utility to terminate.
- If the FILE command was issued from the terminal, an EXIT command in the file causes the utility to return to accepting input from the terminal.
- If a file is currently executing as a result of another FILE command, an EXIT command in that file returns control to the previous file. The utility does not run any commands in a script beyond the EXIT command. EXIT causes an immediate return to the "calling" layer.

## Syntax

```
EXIT
```

## Example

```
EXIT
```

The utility terminates.

## FILE

The FILE command causes the current mode of input to be suspended and redirects the scripting utility to accept input from the specified file. Either the end of the file or an EXIT command in the specified file causes the utility to again accept input from the previous input source.

FILE commands can be nested. Nesting FILE commands means that a file being executed through a FILE command can have FILE commands within its command set. The only limitation on how deep FILE commands may be nested is based on the host array's resources.

## Syntax

```
FILE <filename>
```

An extension is not required by the utility, although you can choose one appropriate to your environment.

## Examples

```
FILE snapd1.txt
```

This example executes the file `snapd1.txt` from the current directory.

```
FILE d:\scripts\snapshots\d27.txt
```

This example executes a file `d27.txt` located in the specified directory.

## HELP

The HELP command displays information about using the utility help.

To get HELP on specific command syntax, enter a space and a question mark (?) wherever you would normally include a parameter, at any level of a command line. The help system will then list the options available for that parameter.

## Syntax

```
HELP
```

## Examples

```
HELP
```

Command syntax help is available at any point within a command by typing a '?'

This example accesses HELP and requests the display of the different valid values for the switch REDUNDANCY.

```
Add VDISK my_Vdisk REDUNDANCY= ?
```

The system displays:

The options are:

```
VRAID0
```

```
VRAID1
```

```
VRAID5
```



# LS



## NOTE:

With this version of the utility, the LS commands have replaced the SHOW commands. The SHOW commands are still available; however, they are deprecated.

The LS commands display information about various devices in the currently selected array. If you specify the name of an object, the LS command will display detailed information on the object.

The following object types are available for all of the LS commands (except where indicated): CONTAINER, CONTROLLER, DISK, DISK\_GROUP, DISKSHELF, DR\_GROUP, FOLDER, HOST, LUN, SYSTEM, VDISK, MANAGER, OPTIONS, SNAPSHOT, TIME, AND WORLD\_WIDE\_NAME..

The following list shows how you can use the LS command:

- `LS <object type> FULL`  
Lists all the information available for all the objects of that type.
- `LS <object type> FULL XML`  
Lists all the information available for all the objects of that type in XML format.
- `LS <object type> NOFULL` (This is the default and is a no-operation switch.)
- `LS <object type> <object name>XML` — displays the output in extensible markup language (XML).

## Examples

### Example 1

```
LS VDISK FULL
```

This example provides all information about all virtual disks.

### Example 2

```
LS VDISK <name>
```

This example displays all the properties of a particular disk.

## LS CONTAINER

Use the LS CONTAINER command to display information for the containers available on the currently selected array.

### Syntax

```
LS CONTAINER <container name>
```

## LS CONTROLLER

Use the LS CONTROLLER command to display disk configuration information for the controllers connected to the currently selected array.

### Syntax

```
LS CONTROLLER <controller name>
```

## LS DISK

Use the LS DISK command to display disk configuration information for physical disks connected to the currently selected array.

### Syntax

```
LS DISK <disk_name>
```

## LS DISK\_GROUP

The LS DISK\_GROUP command displays the disk groups currently configured by this array. If <disk\_group\_name> is specified, detailed information about the specified disk group is displayed.

### Syntax

```
LS DISK_GROUP <disk_group_name>
```

## LS DR\_GROUP

Use the LS DR\_GROUP command to display a list of data replication groups for the currently selected array. If you specify the name of the DR Group, the command displays detailed information about the specified DR Group.

### Syntax

```
LS DR_GROUP <dr group name>
```

## LS DISKSHELF

Use the LS DISKSHELF command to display information for the disk enclosures connected to the currently selected array.

### Syntax

```
LS DISKSHELF <diskshelf name>
```

## LS FOLDER

The LS FOLDER command displays a list of folders for the currently selected array.

### Syntax

```
LS FOLDER <folder name>
```

This example lists the folders available on the array.

## LS HOST

The LS HOST command displays the hosts currently configured on this array.

### Syntax

```
LS HOST <host_name>
```

## LS LUN

The LS LUN command displays the LUNs currently configured on this array.



---

**NOTE:**

Aliases cannot be used for LUNs.

---

### Syntax

```
LS LUN <LUN name>
```

## LS MANAGER

The LS MANAGER command displays information about the currently selected manager.



---

**NOTE:**

This command does not have an XML output option.

---

### Syntax

```
LS MANAGER
```

### Switch

The switches FULL and NOFULL can be specified but are ignored. Specifying a manager name is also ignored. The LS MANAGER command always displays the information for the currently selected manager.

## LS OPTIONS

The LS OPTIONS command displays the current utility options.



---

**NOTE:**

This command does not have an XML output option.

---

### Syntax

```
LS OPTIONS
```

## LS SNAPSHOT

Use the LS SNAPSHOT command to display the snapshots.

### Syntax

```
LS SNAPSHOT <snap name>
```

## LS SYSTEM

Use the LS SYSTEM command to display the arrays currently managed by HP Command View EVA.

## Syntax

```
LS SYSTEM <system_name>
```

## LS TIME

Use the LS TIME command to display the current time for the selected array.



---

**NOTE:**

This command does not have an XML output option.

---

## Syntax

```
LS TIME
```

## LS VDISK

The LS VDISK command displays the storage currently configured on this array.

## Syntax

```
LS VDISK <vdisk name>
```

## LS WORLD\_WIDE\_NAME

The LS WORLD\_WIDE\_NAME command displays the host World Wide Names (WWNs) that are visible to this array and not already assigned to a host.



---

**NOTE:**

This command does not have an XML output option.

---

## Syntax

```
LS WORLD_WIDE_NAME
```

## REDISCOVER

This command tells HP Command View EVA to conduct rediscovery to find new arrays or update the status of existing arrays. For example, when communication has been lost and then restored, REDISCOVER could be used.

## Syntax

```
REDISCOVER
```

## RESTART

Use the RESTART command to restart a controller shown by the LS CONTROLLER command.

## Syntax

```
RESTART <controller_name>
```

## Switches

The following switches are available for the RESTART command:

### ALL\_PEERS

Restarts all peer controllers (both controllers) on this array.

### NOALL\_PEERS

Default. Restarts only the specified controller.

## Examples

```
RESTART "\\Hardware\Rack 1\Enclosure 7\Controller B" ALL_PEERS
```

This example restarts Controller B, and its peer controller.

```
RESTART "\\Hardware\Rack 1\Enclosure 7\Controller A" NOALL_PEERS
```

This example restarts Controller A but not its peer controller.

## SELECT

Use the SELECT commands to select a manager or an array. After you select an array, the prompt changes to reflect the selected array.

## SELECT MANAGER

Use the SELECT MANAGER command to direct the command prompt to a selected HP Command View EVA or EVA server. All configuration commands run on the selected HP Command View EVA or EVA Command View instance and affect its array's configuration. The <manager\_name> can be a server, management server name, or an IP address. Unless the SAN administrator has changed the password, the default password is "administrator" and the default user is "administrator." The user name and password for the manager are validated in the background on every command.

## Syntax

```
SELECT MANAGER <manager_name> USERNAME=<username>  
PASSWORD=<password>
```

## Switches

The following switches are available for the SELECT MANAGER:

### PASSWORD=

Password associated with the user name that you use to access the HP Command View EVA.

### USERNAME=

User name that you use to access the HP Command View EVA.

## Required switches

The following switches are required with the SELECT MANAGER command:

- PASSWORD=
- USERNAME=

## Example

```
SELECT MANAGER north_campus USERNAME=XXX PASSWORD=XXXX
```

This example selects the manager `north_campus`.

## SELECT SYSTEM

Use the SELECT SYSTEM command to direct the command prompt to the selected array or manager. All configuration commands then affect the selected array. If the system name has spaces in it, the name must be enclosed in double quotes ("").

## Syntax

```
SELECT SYSTEM <system_name>
```

## Examples

```
SELECT SYSTEM employees
```

In this example, the `employees` array is selected.

```
SELECT SYSTEM "payroll storage"
```

In this example, the array `payroll storage` is selected.

## SET

Use the SET commands to change the properties of a: disk, disk group, DR group, folder, host, controller, options, system, disk shelf or virtual disk.



---

**NOTE:**

You can perform only one property change per SET command.

---

## SET DR\_GROUP

Use SET DR\_GROUP to set the mode of operation of an existing DR Group.

## Syntax

```
SET DR_GROUP <dr_group_name>
```

## Switches

The switches available with the SET DR\_GROUP command are described in the following paragraphs.



---

**NOTE:**

Note that from the destination side, a user can set only the COMMENT and NAME switches.

---

### ACCESSMODE=

The access rights for a connected host. Possible values are READONLY, PRESENTONLY and DISABLE.

### ADD\_VDISK=

Name of a source array virtual disk to be added to the DR Group. A destination virtual disk is automatically created on the destination array.



---

**NOTE:**

If a name is already in use, the name will be rejected and you will be asked to enter a new name.

---

When adding a virtual disk to the DR Group, the following two optional switches can be specified:

- DESTINATION\_DISK\_GROUP= The disk group name on the destination array on which the virtual disk is created. This switch can be specified only after the ADD\_VDISK switch.
- DESTINATION\_VDISK\_NAME= The name of the virtual disk that is created on the destination array. The default name is the same as that of the source. This optional switch can be specified only after the ADD\_VDISK switch.
- TARGETREDUNDANCY= The redundancy level of the virtual disk that is created on the destination array.

### COMMENT=

An optional field for attaching comments to the DR Group being created. This string must be enclosed in quotes. The maximum number of characters for this switch is 64.

### DELETE\_VDISK=

Name of a source array virtual disk to be removed from the DR Group. The corresponding destination virtual disk on the destination array is removed from the DR Group and deleted from the array.

### DETACH\_VDISK=

Name of a source array virtual disk to be removed from the DR Group. The corresponding destination virtual disk on the destination array is removed from the DR Group but not deleted. The detached destination virtual disk continues to exist as an independent virtual disk.

### FAILOVER\_SUSPEND

The FAILOVER\_SUSPEND command instructs the controller to perform the failover and after the failover is done, suspend the connection between the DR groups.

### FAILOVER RESUME

The FAILOVER\_RESUME command instructs the controller to perform the failover and after the failover is done, resume the connection between the DR groups.

## FAILSAFE

When you use the FAILSAFE switch and the connection between the source and the destination array fails, all writes are immediately halted and no write is reported as complete until either the connection is restored or the NOFAILSAFE switch is set.

## FORCEFULLCOPY

The FORCEFULLCOPY forces the DR Groups member to fully copy the source members to the destination members and ignore the DR log.

## LINK\_DOWN\_AUTOSUSPEND=

When this switch is enabled, the DR replication is automatically suspended if the link between the arrays goes down. It will stay suspended even if the link returns. The possible values are: ENABLED or DISABLED.

## MAX\_LOG\_SIZE=

This is the maximum size for the DR log disk. The value is the number of blocks in 512 bytes. If left blank or set to "0", the controller firmware will calculate the best log size for the space available. Otherwise, you can specify a size, which must be between 136MB and 2TB entered in blocks: 278528 through 4294967296.



---

### NOTE:

If you specify a size that is larger the available disk space, the command will complete successfully, and the DR Group will be created (even though the space is not available). If additional disks are added to the configuration, the controller will allocate more data for the DR log.

---

## NOFAILSAFE

This is the default when you first create the DR Group. When the connection between the source and the destination array fails, writes are directed to the log until the connection is restored. When the connection is restored, the pending destination writes are pushed across to the destination array. If the log becomes full, the array makes a full copy of the source virtual disk to the destination virtual disk when the connection is restored. Otherwise, the log resynchs the source and destination virtual disks.

## NAME=

This is the new name for the specified DR Group.

## SUSPEND

This option stops replication of data from the source to the destination. This causes the same behavior as if the connection failed.

## NOSUSPEND

This option allows replication of data from the source to the destination to resume. This causes the same behavior as if the connection was restored.

## WRITEMODE=

This option defines the I/O interaction between the destination and source side. Possible values are synchronous and asynchronous.



### Example

```
SET DR_GROUP "\\Data Replication\\Transactions" SUSPEND
```

This example suspends replication for the DR Group Transactions.

## SET DISK

The SET DISK command changes the properties of a disk.

### Syntax

```
SET DISK <name>
```

### Switches

The switches available for the SET DISK command are described in the following paragraphs.

#### COMMENT=

This is an optional switch that associates a user-defined comment with the new array. A maximum of 128 characters is allowed and the string must be enclosed in double quotes.

#### NAME=

This is the new name to give to the specified disk.

### Example

```
SET DISK "\\Disk Groups\\Ungrouped Disks\\Disk 005" NAME=5Disk
```

This example renames the disk Disk 005 to 5Disk.

## SET DISK\_GROUP

The SET DISK\_GROUP command changes the specified disk group properties.

### Syntax

```
SET DISK_GROUP <name>
```

### Switches

The switches available with the SET DISK\_GROUP command are described in the following paragraphs.

#### ADD=

This is the number of disk drives to add to the specified disk group.

#### COMMENT=

This is an optional switch that associates a user-defined comment with the disk group. A maximum of 128 characters are allowed and the string must be enclosed in double quotes.

#### DELETE=

The name of the disk drive to be removed from the disk group.

### NAME=

This is the new name to give to the specified disk group.

### OCCUPANCY\_ALARM=

This is the point when a defined percentage of space is used. When this point is reached, an event is generated and sent to the host informing the administrator that the group is becoming full. Do not use the percentage sign (%) after the number.

### SPARE\_POLICY=

This parameter determines the amount, if any, of storage space set aside for use in the event that disks fail. The default is single.

- NONE—reserves no space within a disk group to allow for data reconstruction for failure of disk drives
- SINGLE—reserves space within a disk group to allow for data reconstruction for failure of a single disk drive
- DOUBLE—reserves space within a disk group to allow for data reconstruction for failure of two disk drives



---

**NOTE:**

The space set aside is not in numbers of physical disks. It is the equivalent amount of storage space spread across all disks.

---

### Example

```
SET DISK_GROUP "\Disk Groups\pool" NAME="nuclear secrets"
```

This example renames the disk group pool with the new name nuclear secrets.

## SET FOLDER

The SET FOLDER command changes the specified folder properties.



---

**NOTE:**

you cannot rename root folders.

---

### Syntax

```
SET FOLDER <name>
```

### Switches

The switches available for the SET FOLDER command are described in the following paragraphs.

### COMMENT=

This is an optional switch that associates a user-defined comment with the new system. A maximum of 128 characters is allowed, and the string must be enclosed in double quotes.

NAME=

New name to give to the specified folder.

## Example

```
SET FOLDER "\\Virtual Disks\top_secret" NAME=everyone_knows
```

This example renames the folder named `top_secret` to `everyone_knows`.

## SET HOST

The SET HOST command changes the host parameter values.



---

**NOTE:**

Use the SET HOST command to add additional FCAs to hosts added to the array with the ADD HOST command.

---

## Syntax

The following switches are available with the SET HOST command.

ADD\_WORLD\_WIDE\_NAME=

This adds the World Wide Name of the host port list.

DELETE\_WORLD\_WIDE\_NAME=

This option deletes the World Wide Name of the host port list.

COMMENT=

This is an optional switch that associates a user-defined comment with the new array. A maximum of 128 characters is allowed, and the string must be enclosed in double quotes.

IP=

This option is the network IP address.

NAME=

This is the new name given to the specified host.

OPERATING\_SYSTEM=

This is the type of operating system for the specified host. Choose one of the following operating systems, which are spelled here as the utility expects:

- HPUX
- IBMAIX
- LINUX
- OPEN\_VMS
- SOLARIS
- TRU64
- UNKNOWN
- WINDOWS
- CUSTOM = (You must include the equal sign after CUSTOM, and the value must be 16-digit hexadecimal characters.)

## Examples

```
SET HOST \Hosts\install OPERATING_SYSTEM=OPEN_VMS
```

This example assigns OpenVMS as the operating system type for the host install.

```
SET HOST \Hosts\install ADD_WORLD_WIDE_NAME=1000-0000-C922-36CA
```

This example adds a WWN to the host install.

## SET CONTROLLER

The SET CONTROLLER command changes the specified controller properties.

### Syntax

```
SET CONTROLLER <name>
```

### Switches

The following switches are available with the SET CONTROLLER command:

#### COMMENT=

This is an optional switch that associates a user-defined comment with the controller. A maximum of 128 characters (including spaces) is allowed, and the string must be enclosed in double quotes.

#### NAME=

New name to give to the specified controller.

## SET DISKSHELF

The SET DISKSHELF command changes the specified disk enclosure properties.

### Syntax

```
SET DISKSHELF <diskshelf name>
```

### Switches

The only switch available for the SET DISKSHELF command is COMMENT.

## COMMENT=

This is an optional switch that associates a user-defined comment with the disk enclosure. A maximum of 128 characters (including spaces) is allowed, and the string must be enclosed in double quotes.

## SET OPTIONS

The SET OPTIONS command changes the utility characteristics.



### NOTE:

The options you set with the SET OPTIONS command are in effect for the current session only. Each time you start the utility, the default options are reinstated.

## Syntax

SET OPTIONS

## Switches

The following switches are available with the SET OPTIONS command.

### COMMAND\_DELAY=

The number of seconds (0–300 seconds) to wait between issuing commands when running a script using the FILE command. The default is 10 seconds. This switch has no effect when you are typing commands at the command line.

### NOCOMMAND\_DELAY

Specifies that no wait time occurs between commands issued from within a FILE command.

### DISPLAY\_WIDTH=

Sets the amount of characters displayed on a line for LS commands. The default is 80. If the utility output is to be parsed, it is useful to set a high line width. A long line keeps the lines from wrapping, making the output easier to cut, paste, and parse. Line widths from 70 to 500 may be specified.

### ON\_ERROR=

- **CONTINUE**—In this mode, only the EXIT command causes the scripting utility to halt. This is the default and is recommended when you are manually typing commands.
- **EXIT\_ON\_ERROR**—Any error causes the scripting utility to exit with an error code. This mode is useful when requesting the entire script to halt immediately if errors occur while executing a script. This causes the scripting utility to exit on any kind of error: failed command, syntax error, or ambiguous command.
- **HALT\_ON\_ERROR**—This is similar to EXIT\_ON\_ERROR. Any error causes the scripting utility to halt but not exit until any key is pressed. Then the scripting utility exits with an error code. This allows you to see the error before the window closes on exit.



### NOTE:

When you are entering commands manually, HP does not recommend using the EXIT\_ON\_ERROR or HALT\_ON\_ERROR modes. You should avoid these modes because the utility exits on any kind of error, including a typographical one.

## RETRIES=

The number of minutes to attempt long period retries, such as when the command view EVA service is busy or restarting. Specify between 1–120 minutes (inclusive). The default is four minutes.

## NORETRIES

Specifies that you do not want the scripting utility to retry commands.

## SAFE\_DELETE

Specifies if you want to delete all dependent or related objects before the specified object can be deleted. For example, if you specified a deletion of the virtual disk that had LUNs presented, the DELETE VDISK command would be rejected with a message telling you that all the LUNs presented from this storage must be deleted before the storage can be deleted.

This is the default. You must always delete related objects if SAFE\_DELETE is specified.

## NOSAFE\_DELETE

Allows the deletion of an object even if related objects are present. The behavior of the NOSAFE\_DELETE command depends on the mode of operation (entering commands manually or running commands through a script) in which the utility is running.

If you are entering commands at a terminal, the Are you sure? prompt displays and you must type **Yes** for the deletion to proceed.



---

### NOTE:

If you try to delete a virtual disk that is presented, the NOSAFE\_DELETE switch will not allow the deletion. The NOSAFE\_DELETE switch will allow you to delete an entire array (which may contain presented virtual disks).

You must type **Yes** to proceed.

If you are in a FILE command in a script and you have set NOSAFE DELETE, it will delete the virtual disk without a prompt.

---



---

### CAUTION:

Using the NOSAFE\_DELETE switch can cause accidental deletion of virtual disks or presentations.

---

## Example

```
SET OPTIONS ON_ERROR= HALT_ON_ERROR NORETRIES
```

This example sets the options for the utility to stop upon encountering an error and not to retry the commands.

## SET SYSTEM

Use the SET SYSTEM command to modify array properties.



---

### NOTE:

The SET SYSTEM CONSOLE\_LUN\_ID value is used for IBM AIX (set to zero), OpenVMS (required), and Tru64 UNIX (recommended). Other host operating systems ignore the value. Refer to the host operating system installation guides that came with your host system for more information.

---

## Syntax

```
SET SYSTEM <name>
```

## Switches

The following switches are available for the SET SYSTEM command.

### COMMENT=

This is an optional switch that associates a user-defined comment with the new array. A maximum of 128 characters is allowed, and the string must be enclosed in double quotes.

### CONSOLE\_LUN\_ID=

This is the LUN used for console communication. If set to zero, a console LUN is not presented to the host.

### NAME=

This is the new name to give to the specified array.

### MANAGE

This specifies the array to be managed by the selected manager.



---

**NOTE:**

This option allows switching between multiple instances of HP Command View EVA.

---

## Example

```
SET SYSTEM engineering NAME=accounting
```

This example renames the array engineering to accounting.

## SET VDISK

Sets the parameters of the virtual disk.

## Syntax

```
SET VDISK <vdisk name>
```

## Switches

The following switches are available for the SET VDISK command.

### COMMENT=

This is an optional switch that associates a user-defined comment with the new array. A maximum of 128 characters is allowed and the string must be enclosed in double quotes.

### CHANGE\_INTO\_CONTAINER

This is an optional switch that changes a vdisk back into an empty container.



---

**NOTE:**

Note that the `CHANGE_INTO_CONTAINER` option (`SET VDISK` command) will create an empty container and erase any data in the virtual disk.

---

### MIRRORCACHE=

This sets the controller's mirror cache. The values are `MIRRORED` (cache is mirrored between both controllers) and `NOTMIRRORED` (cache is not mirrored).



---

**NOTE:**

For active-active controllers, the only option is `MIRRORED`. If you try to set to `NOTMIRRORED`, you will get an error.

---

### NAME=

This is the new name to give to the specified virtual disk.

### OS\_UNIT\_ID=

This is the ID that is presented to the host operating system. If set to zero, an ID is not presented to the host.

### PREFERRED\_PATH=

This is the preferred controller path that is specified to handle all I/O for the virtual disks. If a controller fails, the path always reverts to the working controller.

- `PATH_A_BOTH`—Controller path A fails over to controller B. When controller A restarts, the virtual disks fail back to controller A. This is failover/failback mode.
- `PATH_A_FAILOVER`—Controller path A fails over to controller B. When controller A restarts, the virtual disks do not fail back over to controller A. This is failover-only mode.
- `PATH_B_BOTH`—Controller path B fails over to controller A. When controller B restarts, the virtual disks fail back to controller B. This is failover/failback mode.
- `PATH_B_FAILOVER`—Controller path B fails over to controller A. When controller B restarts, the virtual disks do not fail back over to controller B. This is failover-only mode.

### NOPREFERRED\_PATH

This allows the I/O to be handled by any controller.

### READ\_CACHE

This specifies the reads are satisfied from the controller's cache.

### NOREAD\_CACHE

This specifies the reads are always satisfied from media, not the controller's cache.

### SIZE=

This indicates the new larger size (you cannot change to a size smaller than the original) of the storage in gigabytes. You can specify size in whole gigabytes only. Fractions are not allowed. The value of `SIZE` can be 2 GB through 2000 GB depending on the disk group's available space.



## WORLD\_WIDE\_LUN\_NAME=

This sets the World Wide LUN Name on unpresented storage (either ACTIVE or snapshot). This switch is rejected when issued against storage that is presented to a host.



### NOTE:

This switch is commonly used to allow a host to point to a new version of a snapshot by giving the new snapshot the same WWN as the old snapshot.

## WRITE\_PROTECT

This does not allow writing to the virtual disk for all presented LUNs.

## WRITE\_CACHE=

This is the cache parameter for the write operation. There are two values: WRITETHROUGH and WRITEBACK. With WRITETHROUGH the operation completes when the write to disk completes. With WRITEBACK, the operation completes when the data is written to cache.

## NOWRITE\_PROTECT

This allows writing to the virtual disk for all presented LUNs.

### Example

```
SET VDISK "\\Virtual Disks\\archive\\ACTIVE"  
WORLD_WIDE_LUN_NAME=6000-1fe1-ff00-0000
```

This example assigns the World Wide LUN Name to the ACTIVE virtual disk of the archive family.

## SHUTDOWN

Use the SHUTDOWN command to shut down any controller displayed by the LS CONTROLLER command.

### Syntax

```
SHUTDOWN <controller_name>
```

### Switches

The following switches are available with the SHUTDOWN command.

- ALL\_PEERS  
Shuts down all peer controllers (both controllers) on this array. This will also power off all disk enclosures.
- NOALL\_PEERS  
Default. Shuts down only the specified controller.

### Examples

```
SHUTDOWN "\\Hardware\\Rack 1\\Enclosure 7\\Controller A"
```

This example shuts down only Controller A.

```
NOALL_PEERS
```

```
SHUTDOWN "\Hardware\Rack 1\Enclosure 7\Controller B" ALL_PEERS
```

This example shuts down Controller B and the peer.

---

## 3 Configuration examples

This chapter provides an example that describe how to use commands to create a simple configuration and to view specific array information.

Select a management server `cvevaserver` to manage using the user default username and password (administrator/administrator):

```
NoSystemSelected: select manager cvevaserver username=administrator
password=administrator
```

List the Systems (use the `LS SYSTEM` command to list arrays):

```
NoSystemSelected: ls system Systems available on this Manager:
  Uninitialized Storage System [5000-1FE1-0015-1F50]
```

Select an array using the `SELECT SYSTEM` command (if an array is uninitialized, it will be shown "Uninitialized Storage System" with the world wide name `xxxx-xxxx-xxxx-xxxx`):

```
NoSystemSelected: select system "Uninitialized Storage System [5000-1FE1-0015-1F50]"
```

Add an array "Archive" (use the help system (?) to show available switches):

```
Uninitialized Storage System [5000-1FE1-0015-1F50]: add system Archive ?
  The options are:
    COMMENT
    CONSOLE_LUN_ID
    DEVICE_COUNT
    DISKGROUP_DISKTYPE
    SPARE_POLICY
```

Add 8 disks to the Archive array:

```
Uninitialized Storage System [5000-1FE1-0015-1F50]: add system Archive device_count=8
```

Use the `LS SYSTEM` command to show arrays:

```
NoSystemSelected: ls system
Systems available on this Manager:
  Archive
```

Select the array Archive:

```
NoSystemSelected: select system Archive
```

Add a 10GB virtual disk called History to Archive:

```
Archive: add vdisk History size=10
```

Add a new host called MainServer (use the help system (?) to list switches for ADD command):

```
Archive: add host MainServer ?
  The options are:
```

```
COMMENT
IP
OPERATING_SYSTEM
WORLD_WIDE_NAME
```

Add MainServer to the array using WWN and specifying the operating system (hpux):

```
Archive: add host MainServer world_wide_name=1234-4321-1234-4231 operating_system=hpux
```

Present virtual disk History as LUN3 to MainServer:

```
Archive: add lun 3 vdisk=History host=MainServer
```

Use the LS command to list virtual disks available on MainServer:

```
Archive: ls vdisk
```

Vdisks available on this Cell:

```
\Virtual Disks\History\ACTIVE
```

Use the LS command to view the virtual disk History:

```
Archive: ls vdisk History
```

```
\Virtual Disks\History\ACTIVE information:
```

object

```
objectid .....: 00200710b4080560ae420100005002000000c700
objectname .....: \Virtual Disks\History\ACTIVE
objecttype .....: virtualdisk
objecthexuid .....: 6005-08b4-0001-42ae-0002-5000-00c7-0000
partitionname .....: ACTIVE
uid .....: 8192.7.16.1610942644.82606.151552.13041664
objectparentuid .....: 0.7.16.1610942644.82606.151552.12976128
objectparenthexuid .....: 6005-08b4-0001-42ae-0002-5000-00c6-0000
objectparentid .....: 00000710b4080560ae420100005002000000c600
creationdatetime .....: 01-Feb-2005 17:43:30
timestampmodify .....: 687417000
familyname .....: History
wwlunid .....: 6005-08b4-0001-42ae-0002-5000-00c7-0000
operationalstate .....: good
operationalstatedetail .....: operating_normally
allocatedcapacity .....: 10
virtualdisktype .....: original
requestedcapacity .....: 10
sharingrelationship .....: none
sharinginformation
  parentvdiskhexuid .....: n/a
  parentvdiskid .....: n/a
```

```

parentvdiskname .....: n/a
childvdiskhexuid .....: n/a
childvdiskid .....: n/a
childvdiskname .....: n/a
redundancy .....: vraid1
writecache .....: writeback
vdisksecondarystate .....: none
mirrorcache .....: mirrored
readcache .....: enable
virtualdiskpresented .....: yes
presentations
  presentation
    hostid .....: 00800710b4080560ae420100005002000000cb00
    lunnumber .....: 3
    hostname .....: \Hosts\MainServer
writeprotect .....: disable
osunitid .....: 0
diskgroupname .....: \Disk Groups\Default Disk Group
diskgroupid .....: 00010710b4080560ae420100005002000000b400
preferredpath .....: no_preference

```

Use the LS command to list the available hosts on MainServer:

Archive: ls host MainServer

Hosts available on this Cell:

\Hosts\MainServer information:

object

```

objectid .....: 00800710b4080560ae420100005002000000cb00
objectname .....: \Hosts\MainServer
objecttype .....: host
objecthexuid .....: 6005-08b4-0001-42ae-0002-5000-00cb-0000
hostname .....: MainServer
uid .....: 32768.7.16.1610942644.82606.151552.13303808
objectparentuid .....: 1028.4.4.67372036.67372036.67372036.67372036
objectparenthexuid .....: 0404-0404-0404-0404-0404-0404-0404-0404
ipaddress .....: dynamic_ip_assignment
presentations
  presentation
    lunnumber .....: 3
    virtualdiskid .....: 00200710b4080560ae420100005002000000c700
    virtualdiskname .....: \Virtual Disks\History\ACTIVE

```

```

operationalstate .....: good
operationalstatedetail .....: initialized_ok
fcadapterports
  port
    portwwn .....: 1234-4321-1234-4231
directeventing .....: enable
osmode .....: hpux
osmodebitmask .....: n/a

```

Use the LS command to list the available LUNs on MainServer storage cell 3:

```
Archive: ls lun \Hosts\MainServer\3
```

LUNs available on this Cell:

\Hosts\MainServer\3 information:

object

```

objectid .....: 00880710b4080560ae420100005002000000cc00
objectname .....: \Hosts\MainServer\3
objecttype .....: presentedunit
objecthexuid .....: 6005-08b4-0001-42ae-0002-5000-00cc-0000
virtualdiskname .....: \Virtual Disks\History\ACTIVE
virtualdiskid .....: 00200710b4080560ae420100005002000000c700
hostname .....: \Hosts\MainServer
hostid .....: 00800710b4080560ae420100005002000000cb00
lunnumber .....: 3

```

Use the LS command to view the LUNs on \Hosts\MainServer\3 with XML option to format output in XML:

```
Archive: ls lun \Hosts\MainServer\3 xml
```

\Hosts\MainServer\3 information:

<object>

```

<objectid>00880710b4080560ae420100005002000000cc00</objectid>
<objectname>\Hosts\MainServer\3</objectname>
<objecttype>presentedunit</objecttype>
<objecthexuid>6005-08b4-0001-42ae-0002-5000-00cc-0000</objecthexuid>
<objectdiskname>\Virtual Disks\History\ACTIVE</objectdiskname>
<objectdiskid>00200710b4080560ae420100005002000000c700</objectdiskid>
<hostname>\Hosts\MainServer</hostname>
<hostid>00800710b4080560ae420100005002000000cb00</hostid>
<lunnumber>3</lunnumber>

```

</object>